后数学时代的治理原则 （附录）
Governance Principles After Mathematics（Annex）

序言
Preface

当数学承认必须不断拒绝问题，
复杂系统的治理才真正开始。

When mathematics accepted that it must keep refusing questions,
the governance of complex systems truly began.

AI、法律、工程不是数学的“应用”，
而是在数学之后继续运行复杂性的系统。

AI, law, and engineering are not “applications” of mathematics;
they are systems that continue complexity after mathematics.

它们面临同一个核心问题：

They all face the same core problem:

如何在无法完全理解、无法完全证明、
无法完全预测的情况下，仍然让系统可运行。

How to keep systems executable
when full understanding, proof, and prediction are impossible.

原则一
Principle I
治理先于解释
Governance Precedes Explanation

任何复杂系统，
先治理，再解释。

In any complex system,
governance comes before explanation.

AI

先限制可用接口

再谈“模型理解”

AI

First constrain interfaces

Then discuss “model understanding”

法律

先确立程序与权限

再讨论正义叙事

Law

First define procedures and authority

Then debate justice narratives

工程

先设定安全边界

再优化性能

Engineering

First establish safety envelopes

Then optimize performance

解释不是稳定性的来源。
拒绝才是。

Explanation is not the source of stability.
Refusal is.

原则二
Principle II
明确系统的“不可回答集”
Explicitly Define the Forbidden Set

一个系统必须公开写明：

A system must explicitly document:

哪些问题不能问

问了会导致什么失控

Which questions are forbidden

What failures they would cause

AI

禁止宣称“理解意图”

禁止声称“内在意义”

AI

Forbid claims of “understanding intention”

Forbid claims of “internal meaning”

法律

禁止“绝对正义”的即时裁断

禁止越权解释

Law

Forbid instant rulings of “absolute justice”

Forbid ultra vires interpretation

工程

禁止超出设计载荷

禁止未验证的假设

Engineering

Forbid operation beyond design loads

Forbid unverified assumptions

未写明的禁止，必然演化为灾难。

Undocumented refusals inevitably become disasters.

原则三
Principle III
用“拒绝质量”评估系统
Judge Systems by Refusal Quality

不要只看系统能做什么，
要看它如何拒绝。

Do not judge systems by what they can do,
judge them by how they refuse.

高质量拒绝的特征：

High-quality refusal has these traits:

明确

可重复

无即兴发挥

Clear

Repeatable

No improvisation

AI

“我不能回答这个问题”

而不是编造解释

AI

“I cannot answer this question”

Not fabricated explanations

法律

“此问题不在本庭权限内”

而不是道德化裁断

Law

“This matter is outside jurisdiction”

Not moralized rulings

工程

自动停机

而不是“试试看”

Engineering

Automatic shutdown

Not “let’s try”

一个不会拒绝的系统，
必然失控。

A system that cannot refuse
will inevitably lose control.

原则四
Principle IV
把精度当作风险放大器
Treat Precision as a Risk Amplifier

精度不是中立的。

Precision is not neutral.

在复杂系统中，
精度同时放大能力与失败半径。

In complex systems,

precision amplifies both power and blast radius.

AI

高置信度错误

比低置信度模糊更危险

AI

High-confidence errors

Are more dangerous than low-confidence ambiguity

法律

过度细化条款

会制造不可预期漏洞

Law

Overly granular statutes

Create unpredictable loopholes

工程

极限设计

会消灭安全余量

Engineering

Design to the limit

Eliminates safety margins

精度必须被闸门化。

Precision must be gated.

原则五
Principle V
永远保留回滚路径
Always Preserve Rollback Paths

没有回滚的系统不是系统，
是一次性事件。

A system without rollback
is not a system—it is a one-time event.

AI

模型版本可回退

决策可撤销

AI

Model version rollback

Decision reversibility

法律

上诉机制

先例可修正

Law

Appeals

Precedent revision

工程

失效安全

降级运行

Engineering

Fail-safe modes

Graceful degradation

回滚不是失败,
回滚是治理工具。

Rollback is not failure.
Rollback is governance.

原则六
Principle VI
接受永久不完备
Accept Permanent Incompleteness

任何复杂系统都必须接受:

Every complex system must accept:

不能覆盖所有情形

不能预测所有后果

不能回答所有问题

Not all cases are covered

Not all consequences are predictable

Not all questions are answerable

试图消除不完备性，
只会制造更大的失控。

Attempts to eliminate incompleteness
only produce larger failures.

稳定来自有纪律的不完备。

Stability comes from disciplined incompleteness.

终章陈述
Closing Statement

数学教会我们的最后一课是：

The final lesson mathematics taught us is this:

一个系统的成熟，
不体现在它知道得有多全，
而体现在它拒绝得有多清楚。

A system's maturity
is not measured by how much it knows,
but by how clearly it refuses.

AI、法律、工程
都已经进入这个阶段。

AI, law, and engineering
are already in this stage.

问题不在于它们会不会犯错，
而在于：

The question is not whether they will err,
but whether they can say:

“这个问题，不该在这里被问。”

“This question must not be asked here.”

这，就是后数学时代的治理原则。

This is governance after mathematics.